2次の効用関数を持つ不可分財の最適配分問題に対する近似解法の研究 (計算機科学とアルゴリズムの数理的基礎とその応用)

鈴木, 瞬也; 塩浦, 昭義

数理解析研究所講究録
第 1744 巻 2011 年 85-92

2次の効用関数を持つ不可分財の最適配分問題に対する近似解法の研究

鈴木瞬也* 塩浦昭義†

1 はじめに

1.1 研究の概要

本研究では, 不可分財に対する**組合せオークション**から生じる**財の配分問題**について考える. 組合せオークションでは, 複数の不可分財を複数の参加者に配分する. 各参加者には, 財の部分集合に対する満足度を表す効用関数が与えられているとする. このとき, オークション主催者としての自然な目的は, オークションの社会的余剰 (社会的厚生), すなわち, オークション参加者全員の効用関数値の総和を最大化することである.

本研究で扱う財の配分問題を数学的に記述する. 組合せオークションでは, m 個の異なる不可分財を n 人の参加者に配分するものとする. 配分される財の集合を $M=\{1,2,\ldots,m\}$, 参加者の集合を $N=\{1,2,\ldots,n\}$ とおき, 各参加者 $k\in N$ の配分された財の集合に対する満足度を表す効用関数を $v_k:2^M\rightarrow\mathbf{R}$ とおくと, 財の配分問題は次のように定式化される.

財の配分問題

- 最大化: $\sum_{k\in N} v_k(S_k)$
- 条件: $S_k\subseteq M\ \ (k\in N),\ \ \cup_{k\in N}S_k=M$, $S_k\cap S_l=\emptyset\qquad(\forall k,l\in N,k\neq l)$

このような一般的な財の配分問題において, 効用関数を陽に表現するのに必要な記憶容量が指数サイズになってしまうという問題がある. この問題に対処するため, 本研究では, **2次形式の効用関数**について考える.

集合関数 $v:2^M\rightarrow\mathbf{R}$ は, 実数 $w(i)\ \ (i\in M)$ および $w(i,j)\ \ (i,j\in M,i<j)$ を用いて

$$v(S)=\sum_{i\in S}w(i)+\sum_{i,j\in S,i<j}w(i,j)$$

と表されるとき, 2次形式と呼ばれる. ただし, $w(i)$ は任意の財 i に対する満足度を表し, $w(i,j)$ は任意の2種類の財 i,j に対する満足度の追加量を表すものとする. 2次の効用関数を用いると, 記憶容量のサイズが $O(m^2)$ に抑えられるという利点がある. 組合せオークションにおける2次の効用関数は Conitzer らによって提案され [2], 一般の2次の効用関数に対する財の最適配分問題が NP 困難であることが示されている.

本研究では, 2次の効用関数が**優モジュラ関数**の場合, **劣モジュラ関数**の場合に分けて財の配分問題を考える. 優モジュラ関数は互いの性質を補う2種類の財の関係である**補完性**, 劣モジュラ関数は似た性質を持つ2種類の財の関係である**代替性**と等価である.

1.2 研究の背景

本研究の背景として, 組合せオークションが近年注目を浴びているということが挙げられる. 一つのきっかけとなったのは, FCC (アメリカ連邦通信委員会) による周波数帯域割り当てが許可制からオークションによる販売に切り替わったということである. そのオークションでは, 周波数帯を個別に競りにかけるのではなく, いくつかの周波数帯をまとめて競りにかけていた.

周波数オークションに限らず, 組合せオークションは空港での離発着権の割り当て, トラックの業者間

*東北大学大学院情報科学研究科
†第1著者に同じ

での配送の請負等に適用されている．今後の応用例としては，電子商取引への適用，旅行販売や不動産販売への適用が挙げられる．

また，組合せオークションは他の多くの複雑な組合せ最適化問題の近似に適用されることがある．例として，チリにおける給食の配分効率化問題への適用が挙げられる [3]．

1.3 既存研究

財の配分問題において，効用関数を陽に表現するのに必要な記憶容量が指数サイズになってしまうため，過去の研究では効用関数を何らかのオラクルによって暗に表現することが一般的である．良く用いられるオラクルとしては，効用関数の関数値を評価する value オラクルがある．このほか，効用関数に関するある種の最適化問題の最適解を求めるという，より強力な demand オラクルも用いられることもある．

一般の効用関数に対し，財の配分問題は，効用関数が多項式サイズで陽に与えられた場合であっても，NP 困難であることが知られている．近似アルゴリズムについては，value オラクルを用いた場合では $O(\frac{1}{\sqrt{m}})$ 近似が可能であり [1],[9]，demand オラクルの場合は $O(\frac{\sqrt{\log m}}{m})$ 近似が可能である [6]．

財の配分問題では，応用例が豊富なことから効用関数が劣モジュラ性を持つ場合についても数多くの結果がある．財の配分問題は，効用関数に劣モジュラ性を仮定しても依然として NP 困難であることが知られている．この問題に対し，value オラクルを用いた場合に対する 0.5 近似アルゴリズムが提案されている [8]．一方，value オラクルを用いた場合には $(1-1/e)$ より良い近似比を持つ解を多項式時間で求めることは NP 困難であることが示されている．これに対し，最良の近似比である $(1-1/e)$ 近似解を求める近似アルゴリズムが最近提案された [11]．demand オラクルを用いた場合については，$(1-1/e)$ よりわずかに良い近似比を持つ近似アルゴリズムが提案されている [4]．

1.4 本研究の結果

本研究の結果を表 1 にまとめた．効用関数がすべて 2 次の優モジュラ関数で，参加者が 2 人の場合は，財の配分問題を有向グラフの最小カット問題に帰着することができるので，多項式時間 ($O(m^3/\log m)$) で解くことができることを示す．参加者が 3 人以上の場合は財の配分問題は NP 困難であるため，参加者が 2 人の場合の多項式時間アルゴリズムを利用したヒューリスティックアルゴリズムを提案する．また，LP ラベリングというアルゴリズムを用いると，効用関数がすべて 2 次の優モジュラ関数のとき，参加者の数に関係なく，得られる近似解のもつ近似比の期待値が 0.5 以上となることを示す．効用関数がすべて 2 次の劣モジュラ関数のときは NP 困難であるが，参加者が 2 人の場合は財の配分問題を最大カット問題に帰着することによって，0.874 近似解が得られることを示す．

表 1: 本研究の結果

<table>
<tr><th>効用関数</th><th>参加者 2人</th><th>参加者 3人以上</th></tr>
<tr><td rowspan="2">2次の優モジュラ関数
($w(i,j) \geq 0$)</td><td>$O(m^3/\log m)$
アルゴリズム
（最小 s-t カット問題に帰着）</td><td>・NP困難
（k 端子カット問題を帰着）
・ヒューリスティックアルゴリズムの提案</td></tr>
<tr><td colspan="2">0.5近似アルゴリズムの提案</td></tr>
<tr><td rowspan="2">2次の劣モジュラ関数
($w(i,j) \leq 0$)</td><td colspan="2">NP困難（最大 s-t カット問題を帰着）</td></tr>
<tr><td>0.874近似アルゴリズム
（最大 s-t カット問題に帰着）</td><td></td></tr>
</table>

本稿では，第 2 節で 2 次の優モジュラ関数の場合の 0.5 近似アルゴリズムについて述べる．第 3 節では 2 次の劣モジュラ関数で，参加者が 2 人の場合の 0.874 近似アルゴリズムについて述べる．

2 効用関数について

本研究で扱う優モジュラ関数と劣モジュラ関数という 2 種類の効用関数について説明する．優モジュラ関数は数理経済学における補完性，劣モジュラ関数は代替性と等価である．

集合関数 $v : 2^M \to \mathbf{R}$ は次の不等式 (1) を満たすとき，優モジュラ関数と呼ばれる．

$$v(A) + v(B) \leq v(A \cup B) + v(A \cap B) \quad (\forall A, B \subseteq M) \qquad (1)$$

不等式 (1) は以下の不等式 (2) と等価である.

$$v(A \cup \{x\}) - v(A) \leq v(B \cup \{x\}) - v(B) \quad (\forall A \subseteq \forall B \subseteq M, \forall x \in M - B) \tag{2}$$

定理 1. 2 次形式の効用関数 $v(S) = \sum_{i \in S} w(i) + \sum_{i,j \in S, i<j} w(i,j) \quad (S \subseteq M)$ が優モジュラ関数であることの必要十分条件は, $w(i,j) \geq 0 \quad (\forall i,j \in M, i \leq j)$ である.

集合関数 $v : 2^M \to \mathbf{R}$ は次の不等式 (3) を満たすとき, 劣モジュラ関数と呼ばれる. 劣モジュラ関数は優モジュラ関数の符号を反転したものである.

$$v(A) + v(B) \geq v(A \cup B) + v(A \cap B) \quad (\forall A, B \subseteq M) \tag{3}$$

不等式 (3) は以下の不等式 (4) と等価である.

$$v(A \cup \{x\}) - v(A) \geq v(B \cup \{x\}) - v(B) \quad (\forall A \subseteq \forall B \subseteq M, \forall x \in M - B) \tag{4}$$

定理 2. 2 次形式の効用関数 $v(S) = \sum_{i \in S} w(i) + \sum_{i,j \in S, i<j} w(i,j) \quad (S \subseteq M)$ が劣モジュラ関数であることの必要十分条件は, $w(i,j) \leq 0 \quad (\forall i,j \in M, i \leq j)$ である.

3 効用関数が 2 次の優モジュラ関数の場合の 0.5 近似アルゴリズム

効用関数がすべて 2 次の優モジュラ関数の場合の 0.5 近似アルゴリズムを提案する. このアルゴリズムは Langberg らによって提案された, "Multiway Uncut Problem" に対する近似アルゴリズム [7] を改良したものである.

3.1 整数計画問題と LP 緩和

2 次の効用関数の場合の財の配分問題は, 次のような整数計画問題として定式化できる.

整数計画問題

$$
\begin{array}{lll}
\text{最大化} & \displaystyle\sum_{i \in N} \left(\sum_{u \in V} w_u^i x_u^i + \sum_{(u,v) \in E} w_{uv}^i c_{uv}^i \right) & \\
\text{条件} & \displaystyle\sum_{i \in N} x_u^i = 1 & \text{for all } u \in V \\
& c_{uv}^i = \min(x_u^i, x_v^i) & \text{for all } (u,v) \in E \\
& x_u^i, c_{uv}^i \in \{0,1\} & \text{for all } u, v, i
\end{array}
$$

$w_u^i (\geq 0)$ は参加者 i の財 u に対する満足度を表し, $w_{uv}^i (\geq 0)$ は参加者 i の財 u, v に対する満足度の追加量を表す. 財 u が参加者 i に配分されたとき, $x_u^i = 1$ となり, それ以外は $x_u^i = 0$ である. 財 u, v が共に参加者 i に配分されたとき, $c_{uv}^i = 1$ となり, それ以外は $c_{uv}^i = 0$ である. したがって, この整数計画問題の目的関数は財の配分問題の目的関数に等しい.

この整数計画問題に対する LP 緩和を以下に示す. この LP 緩和を次節のアルゴリズムで用いる.

LP 緩和

$$
\begin{array}{lll}
\text{最大化} & \displaystyle\sum_{i \in N} \left(\sum_{u \in V} w_u^i x_u^i + \sum_{(u,v) \in E} w_{uv}^i c_{uv}^i \right) & \\
\text{条件} & \displaystyle\sum_{i \in N} x_u^i = 1 & \text{for all } u \in V \\
& c_{uv}^i = \min(x_u^i, x_v^i) & \text{for all } (u,v) \in E \\
& x_u^i, c_{uv}^i \geq 0 & \text{for all } u, v, i
\end{array}
$$

3.2 LP ラベリング

3.2.1 アルゴリズム

LP ラベリングというアルゴリズムを用いて, 財の配分を求める. そのアルゴリズムの流れを示す.

(STEP1) LP 緩和の最適解を求める.

(STEP2) V のすべての点 (財) が配分されていない状態にする.

(STEP3) 参加者 $i \in \{1, 2, \ldots, n\}$ と $[0,1]$ の範囲の閾値 ρ をランダムに選ぶ.

(STEP4) 未配分の各点 (財) $u \in V$ に対して, $x_u^i \geq \rho$ ならば u を参加者 i に配分する. この時点で未配分の点が存在する場合は, (STEP3) に戻る. すべての点が配分されたらアルゴリズムを終了する.

3.2.2 LP ラベリングの例

LP ラベリングの具体例を示す. 財の個数は 3 個, 参加者は 3 人とした.

まず, (STEP1) で LP 緩和の最適解を求める. 求めった結果, LP 緩和の最適解 x_u^i が表 2 のような値をとったと仮定して以降の流れを説明する. (STEP2) として V のすべての点 (財) が配分されていない状態にし, その後 (STEP3)→(STEP4) の流れを繰り返す. アルゴリズムの実行例を以下に示す. ここで, (STEP3)→(STEP4) の 1 回の操作をラウンドと呼ぶことにする.

表 2: x_u^i の値

x_u^iの値	参加者1	参加者2	参加者3
財1	0.7	0.2	0.1
財2	0.3	0.3	0.4
財3	0	0.5	0.5

(ラウンド 1) [参加者 2, $\rho = 0.4$]
$x_1^2 = 0.2 < \rho, x_2^2 = 0.3 < \rho, x_3^2 = 0.5 \geq \rho$ より, $x_u^i \geq \rho$ の条件を満たしているのは財 3 のみである. したがって, 財 3 を参加者 2 に配分する. 未配分の点が存在するので, (STEP3) に戻る.

(ラウンド 2) [参加者 1, $\rho = 0.8$]
$x_1^1 = 0.7 < \rho, x_2^1 = 0.3 < \rho$ より, 配分される財はない.

(ラウンド 3) [参加者 1, $\rho = 0.3$]
$x_1^1 = 0.7 \geq \rho, x_2^1 = 0.3 \geq \rho$ より, 財 1 と 2 を参加者 1 に配分する. すべての点について配分が行われたので, アルゴリズムは終了する.

3.3 0.5 近似の証明

LP ラベリングを用いたときに得られる近似解のもつ近似比の期待値が 0.5 以上となることを証明する. 証明の前に, 証明中の記号や文字について説明する.

- $u \mapsto i$ ··· 点 u が参加者 i に配分されること
- $u \mapsto *$ ···
  点 u がいずれかの参加者に配分されること
- $X_u^i = \begin{cases} 1 & (\text{LP ラベリングで } u \mapsto i) \\ 0 & (\text{その他}) \end{cases}$
- $X_{uv}^i = \begin{cases} 1 & (u \mapsto i \text{ かつ } v \mapsto i) \\ 0 & (\text{その他}) \end{cases}$
- $C_{uv} = \sum_i c_{uv}^i$

u を未配分の点と仮定する. そのとき, あるラウンドで点 u が参加者 i に配分される確率は次のようになる.

$$\Pr[u \mapsto i \text{ in the round}] = \frac{1}{n} \cdot x_u^i$$

したがって, あるラウンドで点 u がいずれかの参加者に配分される確率は次のようになる.

$$\Pr[u \mapsto * \text{ in the round}] = \frac{1}{n} \cdot \sum_i x_u^i = \frac{1}{n}$$

補題 1. X_u^i の期待値 $\mathrm{E}[X_u^i]$ は, $\mathrm{E}[X_u^i] = x_u^i$ である.

証明: あるラウンド r の前は点 u は未配分で, ラウンド r で点 u が参加者 i に配分される確率を求め, その確率の $r = 1$ から $r = \infty$ までの和を求めればよい. したがって,

$$\begin{aligned} \mathrm{E}[X_u^i] &= \sum_{r=1}^{\infty}(1 - \Pr[u \mapsto * \text{ before round } r]) \\ &\qquad \cdot \Pr[u \mapsto i \text{ in round } r] \\ &= \sum_{r=1}^{\infty}\left(1 - \frac{1}{n}\right)^{r-1} \cdot \frac{x_u^i}{n} \\ &= x_u^i \end{aligned}$$

□

補題 2. X_{uv}^i の期待値 $\mathrm{E}[X_{uv}^i]$ について, $\mathrm{E}[X_{uv}^i] \geq \frac{c_{uv}^i}{2-C_{uv}}$ が成り立つ.

証明: u と v 両方が同じラウンドで参加者 i に配分される確率を求めることによって, $\mathrm{E}[X_{uv}^i]$ の下界を求める.

u と v はどちらも未配分の点であると仮定する.

(a) あるラウンドで u と v が同時に参加者 i に配分される確率

$$\frac{1}{n}\min(x_u^i, x_v^i) = \frac{1}{n}c_{uv}^i$$

(b) あるラウンドで u と v のどちらかがいずれかの参加者に配分される確率

$$\frac{1}{n}\sum_{i=1}^{n}\max(x_u^i, x_v^i) = \frac{1}{n}\cdot(2 - C_{uv})$$

ここで, $\sum_{i=1}^{n}(\min(x_u^i, x_v^i) + \max(x_u^i, x_v^i)) = \sum_{i=1}^{n}(x_u^i + x_v^i) = 2$ より,
$\sum_{i=1}^{n}\max(x_u^i, x_v^i) = 2 - \sum_{i=1}^{n}\min(x_u^i, x_v^i) = 2 - \sum_{i=1}^{n} c_{uv}^i = 2 - C_{uv}$ であることを用いている.

したがって, u と v 両方が同じラウンドで参加者 i に配分される確率は次のようになる.

$$\begin{aligned}
&\Pr[u \text{ and } v \text{ are assigned label } i \\
&\qquad\qquad\qquad \text{in the same round}] \\
&= \sum_{r=1}^{\infty}(1 - \Pr[u \mapsto * \text{ or } v \mapsto * \text{ before round } r]) \\
&\qquad\qquad \cdot\Pr[u \mapsto i \text{ and } v \mapsto i \text{ in round } r] \\
&= \sum_{r=1}^{\infty}\left(1 - \frac{2 - C_{uv}}{n}\right)^{r-1}\cdot\frac{c_{uv}^i}{n} \\
&= \frac{c_{uv}^i}{2 - C_{uv}}
\end{aligned}$$

□

定理 3. LP ラベリングのアルゴリズムは, 0.5 近似アルゴリズムである.

証明: 補題 1 と補題 2 を用いて近似解の期待値を計算する. 近似解の期待値は,

$$\begin{aligned}
&\sum_{i\in N}\left(\sum_{u\in V} w_u^i \mathrm{E}[X_u^i] + \sum_{(u,v)\in E} w_{uv}^i \mathrm{E}[X_{uv}^i]\right) \\
&\geq \sum_{i\in N}\left(\sum_{u\in V} w_u^i x_u^i + \sum_{(u,v)\in E} w_{uv}^i \frac{c_{uv}^i}{2 - C_{uv}}\right) \\
&\geq \sum_{i\in N}\left(\sum_{u\in V} w_u^i x_u^i + \sum_{(u,v)\in E} w_{uv}^i \frac{c_{uv}^i}{2}\right) \\
&\geq 0.5 \times (\text{LP 緩和の最適値}) \\
&\geq 0.5 \times (\text{目的関数の最適値})
\end{aligned}$$

以上から, LP ラベリングのアルゴリズムは 0.5 近似アルゴリズムである. □

4 効用関数が 2 次の劣モジュラ関数で, 参加者が 2 人の場合の近似アルゴリズム

効用関数がすべて 2 次の劣モジュラ関数で, 参加者が 2 人の場合の近似アルゴリズムを提案する. 2 次の劣モジュラ関数のとき, 財の配分問題を有向グラフの最大カット問題に帰着させることによって, 財の配分問題の 0.874 近似解が得られることを示す. 最大カット問題は以下に示した通り, グラフ $G = (V, E)$ のカットの重みを最大化する問題である.

最大 s-t カット問題

- 入力: ソース s, シンク t を持つ有向グラフ $G = (V, E)$ と非負の枝重み $\omega : E \to R^+$
- 出力: V の分割 (S, T) $(T = V - S, s \in S, t \in T$ を満たす$)$
- 目的: カットの重み $\sum_{i\in S, j\in T}\omega(i, j)$ を最大化

4.1 最大カット問題への帰着

4.1.1 グラフの作成方法

財の配分問題が与えられたとき，グラフ G を作成する方法について説明する．グラフ G の頂点集合は $V = M \cup \{s, t\}$，枝集合 E は，以下に定義する枝集合 E_1 と，枝集合 E_2 の和集合となる．枝集合 E_1 は，1 つの財 $(i \in M)$ に対する満足度の値である $w_1(i), w_2(i)$ を用いて定義される．枝集合 E_2 は，財のペア $(i, j \in M, i < j)$ に対する満足度の追加量の値である $w_1(i,j), w_2(i,j)$ を用いて定義される．

(i) 枝集合 E_1 の定義
各財 $i \in M$ に対して，図 1 のようにソース s から点 i の向きに重み $w_2(i)$ の枝を張り，点 i からシンク t の向きに重み $w_1(i)$ の枝を張る．

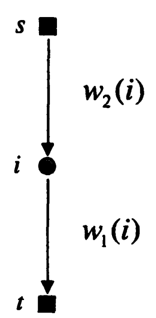

図 1: 枝集合 E_1 の定義

(ii) 枝集合 E_2 の定義
各財のペア $i, j (i < j)$ に対して，図 2 のようにソース s から点 i の向きに重み $w_2(i,j)$ の枝を張り，点 j からシンク t の向きに重み $w_1(i,j)$ の枝を張る．さらに，点 j から点 i の向きに重み $-w_1(i,j) - w_2(i,j)$ の枝を張る．

4.1.2 s-t カットの重みの計算

定義したグラフ $G = (V, E)$ の s-t カットの重みを計算する．枝集合 E_1, E_2 それぞれについてカットの重みを計算する．

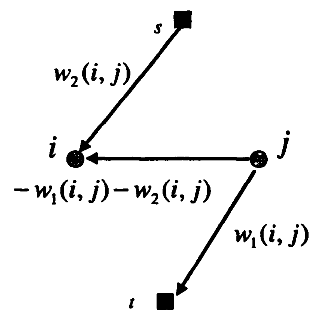

図 2: 枝集合 E_2 の定義

(i) 枝集合 E_1 について
図 1 より，$i \in S$ のとき，重みが $w_1(i)$ である i から t への枝が s-t カットの枝の 1 つとなる．また，$i \in T$ のとき，重みが $w_2(i)$ である s から i への枝が s-t カットの枝の 1 つとなる．

したがって，枝集合 E_1 についての s-t カットの重みは，

$$\sum_{i \in S} w_1(i) + \sum_{i \in T} w_2(i)$$

(ii) 枝集合 E_2 について
枝集合 E_2 について s-t カットの重みを計算する．図 2 より，

$i, j \in S$ のとき，　$w_1(i,j)$
$i, j \in T$ のとき，　$w_2(i,j)$
$i \in S, j \in T$ のとき，　0
$i \in T, j \in S$ のとき，
$w_2(i,j) - w_1(i,j) - w_2(i,j) + w_1(i,j) = 0$

したがって，枝集合 E_2 についての s-t カットの重みは，

$$\sum_{i,j \in S, i<j} w_1(i,j) + \sum_{i,j \in T, i<j} w_2(i,j)$$

枝集合 E は，枝集合 E_1 と枝集合 E_2 の和集合であるから，s-t カットの重みは

$$\sum_{i \in S} w_1(i) + \sum_{i \in T} w_2(i) + \sum_{i,j \in S, i<j} w_1(i,j) + \sum_{i,j \in T, i<j} w_2(i,j)$$
$$= (\text{財の配分 } (S - \{s\}, T - \{t\}) \text{ の目的関数値})$$

となる. したがって, 作成したグラフの最大カットを求めることと, 財の配分の最適解を求めることが等価となる.

4.2 近似アルゴリズム

効用関数がすべて 2 次の劣モジュラ関数で, 参加者が 2 人の場合, 以下に示す近似アルゴリズムによって財の配分問題における近似解を求めることができる.

(STEP1) 4.1.1 節で説明した方法を用いて有向グラフを作成する.

(STEP2) 作成した有向グラフの最大 s-t カットを求める. そのときの集合 $\{S-\{s\}\}$ が参加者 1 への配分, 集合 $\{T-\{t\}\}$ が参加者 2 への配分である.

有向グラフの最大 s-t カットを求める近似アルゴリズムの中で, 既知の最良の近似精度は 0.874 である. 1994 年, Goemans と Williamson が半正定値計画問題 (SDP) を用いた有向グラフの最大カット問題の 0.796 近似アルゴリズム [5] を提案したが, 2002 年に Lewin らによって近似精度が 0.874 に改善されている [10].

したがって, 2 次の劣モジュラ関数で, 参加者が 2 人の場合のアルゴリズムは, 0.874 近似アルゴリズムとなる.

5 結論

本研究では, 不可分財に対する組合せオークションから生じる財の配分問題について扱った. 特に, オークション参加者の配分された財の集合に対する満足度を表す効用関数が 2 次の効用関数となる場合を考え, その効用関数の総和を最大化する問題について考えた. 任意の 2 種類の財の関係が補完性であるとき, 2 次の効用関数は優モジュラ関数となり, 2 種類の財の関係が代替性であるときは, 効用関数は劣モジュラ関数となる. 本研究では, 効用関数が優モジュラ関数の場合, 劣モジュラ関数の場合に分けて, それぞれについて近似解法を提案した.

本稿では, LP ラベリングというアルゴリズムを用いると, 効用関数が 2 次の優モジュラ関数のとき, 参加者の数に関係なく, 得られる近似解のもつ近似比の期待値が 0.5 以上となることを示した. また, 効用関数が 2 次の劣モジュラ関数で, 参加者が 2 人の場合, 財の配分問題を最大カット問題に帰着することによって 0.874 近似解が得られることを示した.

今後の課題としては, 効用関数が 2 次の劣モジュラ関数で, 参加者が 3 人以上の場合の近似アルゴリズムの提案や, 一般の効用関数に対する近似アルゴリズムの提案が挙げられる.

参考文献

[1] L. Blumrosen, N. Nisan. On the computational power of ascending auctions I: demand queries, *Proceedings of ACM Conference on Electronic Commerce*, pp.29-43, 2005.

[2] V. Conitzer, T. Sandholm and P. Santi. Combinatorial auctions with k-wise dependent valuations, *Proceedings of the National Conference on Artificial Intelligence (AAAI)*, pp.248-254, 2005.

[3] R. Epstein, L. Henríquez, J. Catalán, G.Y. Weintraub, C. Martínez and F. Espejo. A combinatorial auction improves school meals in Chile: a case of OR in developing countries, *Proceedings of International Federation of Operational Research Societies*, pp.593-612, 2004.

[4] U. Feige and J. Vondrák. Approximation algorithms for allocation problems: Improving the factor of 1-1/e, *Proceedings of IEEE Symposium on Foundations of Computer Science*, pp.667-676, 2006.

[5] M.X. Goemans and D.P. Williamson. Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming, *Journal of the ACM*, Vol.42, pp.1115-1145, 1995.

[6] R. Holzman, N. Kfir-Dahav, D. Monderer and M. Tennenholtz. Bundling equilibrium in com-

binatorial auctions, *Games and Economic Behavior* Vol.47, pp.104-123, 2004.

[7] M. Langberg, Y. Rabani, and C. Swamy. Approximation algorithms for graph homomorphism problems, *Proceedings of APPROX and RANDOM*, pp.176-187, 2006.

[8] B. Lehmann, D.J. Lehmann and N.Nisan. Combinatorial auctions with decreasing marginal utilities, *Games and Economic Behavior* Vol.55, pp.270-296, 2006.

[9] D. Lehmann, L. O'Callaghan, Y. Shoham. Truth revelation in approximately efficient combinatrial auctions, *Proceedings of ACM Conference on Electronic Commerce*, pp.96-102, 1999.

[10] M. Lewin, D. Livnat, and U. Zwick. Improved rounding techniques for the MAX 2-SAT and MAX DI-CUT problems, *Proceedings of Integer Programming and Combinatorial Optimization*, pp.67-82, 2002.

[11] J. Vondrák. Optimal approximation for the submodular welfare problem in the value oracle model, *Proceedings of ACM Symposium on Theory of Computing*, pp.67-74, 2008.